

Docket No.: P-106

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

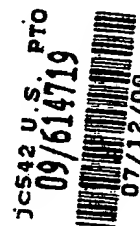
In re Application of

Seung-Hwan BAE

Serial No.: New U.S. Patent Application

Filed: July 12, 2000

For: **DUPLICATING PROCESSORS AND METHOD FOR  
CONTROLLING ANOMALOUS DUAL STATE THEREOF**



TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

Assistant Commissioner of Patents  
Washington, D. C. 20231

Sir:

At the time the above application was filed, priority was claimed based on the following application:

Korean Patent Application No. 28544/1999 filed July 14, 1999.

A copy of each priority application listed above is enclosed.

Respectfully submitted,  
FLESHNER & KIM, LLP

Daniel Y.J. Kim  
Registration No. 36,186

P. O. Box 221200  
Chantilly, Virginia 20153-1200  
703 502-9440  
Date: July 12, 2000  
DYK/kam



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

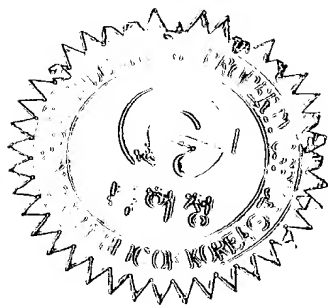
This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Industrial  
Property Office.

출원 번호 : 특허출원 1999년 제 28544 호  
Application Number

출원 년 월 일 : 1999년 07월 14일  
Date of Application

출원인 : 엘지정보통신주식회사  
Applicant(s)

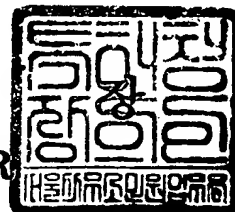
**CERTIFIED COPY OF  
PRIORITY DOCUMENT**



2000      년      06      월      15      일

특      허      청

COMMISSIONER



【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0002
【제출일자】	1999.07.14
【발명의 명칭】	이중화된 프로세서의 이중화 상태 제어 방법
【발명의 영문명칭】	Dual State Control Method Of Duplicated Processors
【출원인】	
【명칭】	엘지정보통신 주식회사
【출원인코드】	1-1998-000286-1
【대리인】	
【성명】	김영철
【대리인코드】	9-1998-000040-3
【포괄위임등록번호】	1999-010680-1
【발명자】	
【성명의 국문표기】	배승환
【성명의 영문표기】	BAE, Seung Hwan
【주민등록번호】	710406-1114224
【우편번호】	780-800
【주소】	경상북도 경주시 안강읍 대동리 68-2
【국적】	KR
【심사청구】	청구
【취지】	특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사를 청구합니다. 대리인 김영철 (인)
【수수료】	
【기본출원료】	19      면                      29,000    원
【가산출원료】	0      면                      0      원
【우선권주장료】	0      건                      0      원
【심사청구료】	6      항                      301,000   원
【합계】	330,000    원
【첨부서류】	1. 요약서·명세서(도면)_1통

**【요약서】****【요약】**

본 발명은 이중화된 프로세서가 비정상적인 이중화 상태로 천이하는 것을 방지하고, 비정상적인 이중화 상태로 천이하더라도 신속하게 복구할 수 있도록 한 이중화된 프로세서의 이중화 상태 제어 방법에 관한 것이다.

종래에는 네트워크를 통해 프로세서를 소프트웨어적으로 이중화 처리하는 경우 해당 네트워크 장애 또는 네트워크 자원의 장애로부터 복구되는 시점에 각 프로세서가 하트비트를 동시에 수신하게 되면, 비정상적인 이중화 상태에 빠져 정상적인 이중화 처리를 수행하지 못하는 문제점이 있었다.

본 발명은 이중화된 프로세서의 시동시에 난수 시드를 다르게 할당하여 난수 시간을 발생시키고, 해당 난수 시간을 사용하여 하트비트 전송 주기를 지속적으로 변경시켜 각 프로세서의 하트비트 수신 시각을 상이하게 하거나, 해당 프로세서의 이중 활성화 상태 발생시 각 프로세서를 기설정된 상태로 천이하게 하거나, 해당 프로세서의 이중화 상태를 지속적으로 감지 및 제어하는 객체를 도입해서 해당 프로세서의 이중화 상태 및 네트워크 자원을 감시하여 이중화 상태 천이를 제어함으로써, 비정상적인 이중화 상태의 발생을 방지할 수 있게 되며, 해당 비정상적인 이중화 상태가 발생하더라도 신속하게 복구할 수 있게 된다.

**【대표도】**

도 3

**【명세서】****【발명의 명칭】**

이중화된 프로세서의 이중화 상태 제어 방법{Dual State Control Method Of Duplicated Processors}

**【도면의 간단한 설명】**

도 1은 일반적인 프로세서에서 이중화 FSM 블록의 상태 천이도.

도 2는 이중화 FSM 블록의 상태 천이를 통한 이중화 처리 동작 순서도.

도 3은 본 발명에 따른 이중 활성화 협상법을 구현하기 위한 동작 순서도.

**【발명의 상세한 설명】****【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

<4> 본 발명은 이중화된 프로세서의 이중화 상태 제어 방법에 관한 것으로, 특히 해당 프로세서가 비정상적인 이중화 상태로 천이하는 것을 방지하고, 비정상적인 이중화 상태로 천이하더라도 신속하게 복구할 수 있도록 한 이중화된 프로세서의 이중화 상태 제어 방법에 관한 것이다.

<5> 종래에는 A측과 B측(물리적인 측면)의 프로세서를 하드웨어적으로 직접 연결하여 이중화(활성화/비활성화) 상태를 구성함에 따라 해당 이중화를 위한 새로운 하드웨어 구조를 설계해야 했으며, 이에 따른 별도의 운영 체제 및 프로그래밍 언어 개발 비용과 시

간이 많이 소요되는 단점이 있었다.

- <6> 이러한 단점을 해결하기 위해 최근의 통신 시스템에서는 각 프로세서를 네트워크로 연결시켜 동기를 제공하고 주기적인 신호인 하트비트(Heartbeat)라는 메카니즘을 제공하여 소프트웨어적으로 이중화를 처리하는데, 이때, 네트워크의 장애 또는 케이블, 허브 등 네트워크 자원의 불량 및 장애가 발생하면 각 프로세서 자신(셀프, self)은 상대방 프로세서(트윈, twin)가 다운된 것으로 판단하게 되며, 특히, 비활성화 측은 트윈이 다운된 것으로 인식하여 활성화 상태로 천이하게 되어 해당 통신 시스템에서 프로세서의 이중 활성화 상태가 발생된다.
- <7> 이 경우 해당 프로세서와 상호 작용하는 외부 네트워크 요소가 이중 활성화된 프로세서들로 인해 동일한 신호나 데이터를 이중으로 수신하게 되어 정상적인 처리를 수행하지 못하게 된다.
- <8> 특히, 네트워크 관련 장애에서 복구되었을 경우 정상적인 이중화 상태로 복구되어야 하지만, 실제로 장애에서 복구되는 시점에 비정상적인 이중 비활성화 상태로 천이하게 되어 외부 네트워크 요소는 해당 프로세서가 없거나 비활성화 상태로 판단하여 외부로의 송신 신호를 발생하지 않는 등의 상호 작용을 수행하지 않게 되는 문제점이 있었다.
- <9> 한편, 통신 시스템에서 각 프로세서는 트윈이 활성화 상태가 아니면 활성화 상태로 천이하는데, 보통 하트비트의 수신 시간이 정확하게 일치하지 않기 때문에 활성화 대기 상태에서 이중 활성화 상태로 천이되는 것을 방지할 수 있었다.
- <10> 즉, 하트비트 수신 간격의 오차로 인해 먼저 활성화 대기 상태에 도달한 측

이 활성화 상태로, 늦게 도달한 측이 비활성화 상태로 각각 천이하여 정상적인 이중화 상태를 유지할 수 있게 된다.

- <11> 한편, 전술한 바와 같은 각 프로세서는 이중화 FSM(Finite State Machine) 블록에서 이중화 처리를 수행하는데, 해당 이중화 FSM 블록의 상태 천이도는 첨부된 도면 도 1과 같으며, 각 상태 천이는 프로세서간에 송수신되는 하트비트에 포함된 이벤트(트윈 시동, 트윈 활성화, 트윈 타임아웃 등과 같은 트윈 상태 이벤트)와 정지명령, 재시동명령, 수동 절체, 초기화 이벤트 등과 같은 외부 이벤트에 의해서만 이루어진다.
- <12> 해당 이중화 FSM 블록의 상태 천이를 통한 이중화 처리 동작을 첨부된 도면 도 2를 참조하여 상세하게 설명하면 다음과 같다.
- <13> 여기서, 이중화 FSM 블록은 수신 하트비트 처리 블록에 의해 포착되는 하트비트에 포함된 트윈의 상태 정보나 절체(switchover) 이벤트에 따라 해당되는 상태 천이 기능을 담당하게 되며, 각 상태마다 대응하는 상태 정보를 발신 하트비트 처리 블록으로 인가하거나, 각 상태 천이시마다 즉각적인 하트비트 신호를 전송하도록 발신 하트비트 처리 블록에 알려주게 된다.
- <14> 그리고, 해당 발신 하트비트 처리 블록은 이중화 FSM 블록으로부터의 상태 정보를 트윈에게 전송하는 기능을 담당하되, 상태 천이시 또는 주기적으로 하트비트를 트윈에게 전송한다.
- <15> 해당 수신 하트비트 처리 블록은 트윈의 하트비트를 포착해 이중화 FSM 블록에서 트윈 상태에 따른 상태 천이를 수행할 수 있도록 해 주는 기능을 담당하되, 타임 아웃이 되는 경우 네트워크 장애이거나 트윈 다운임을 이중화 FSM 블록으로 보고한다.

- <16> 먼저, 이중화 FSM 블록이 구동되면 초기화(Initial) 상태를 유지하게 되고(스텝 S1), 전체 시스템의 각 블록에서 초기화가 종료되어 정상적인 기능을 수행하게 되면 해당 이중화 FSM 블록으로 초기화되었다는 정보를 외부 이벤트인 초기화 이벤트로 전송하게 된다.
- <17> 이때, 해당 이중화 FSM 블록은 이를 수집, 점검하여 시스템의 초기화가 종료되는지를 확인하여(스텝 S2), 해당 초기화가 종료되는 경우 시동(Start) 상태로 천이하게 된다(스텝 S3).
- <18> 해당 시동 상태에서 트윈이 시동 상태인지를 확인하여(스텝 S4), 해당 트윈도 시동 상태인 경우 동시 재시작(Dual Restart)을 시도한 것이므로 어느 측이 활성화 상태로 천이할 것인지를 결정하기 위해 협상(Negotiation) 상태로 천이하게 되고(스텝 S5), 해당 협상 상태에서는 자신이 A측인지를 확인하여(스텝 S6), 자신이 A측이면 활성화 상태로 천이하고(스텝 S7), 자신이 A측이 아니면 즉, 자신이 B측이면 비활성화 상태로 천이하게 된다(스텝 S8).
- <19> 한편, 스텝 S7의 활성화 상태에서 수동 절체(manual Switchover)되는지 그리고, 트윈이 비활성화 상태인지를 확인하여(스텝 S9), 수동 절체되고 트윈이 비활성화 상태인 경우 스텝 S8의 비활성화 상태로 천이하게 된다.
- <20> 그런데, 스텝 S9에서 수동 절체되지 않거나 트윈이 비활성화 상태가 아닌 경우에는 트윈이 활성화 상태인지를 확인하여(스텝 S10), 트윈이 활성화 상태가 아니면 자신은 계속 활성화 상태를 유지하게 되고, 트윈도 활성화 상태이면 즉, 이중 활성화 상태가 되면 수동 절체 이벤트에 의해 자신은 스텝 S8의 비활성화 상태로 천이하게 된다.



- <21> 다른 한편, 스텝 S8의 비활성화 상태에서 수동 절체되는지를 확인하여(스텝 S11), 수동 절체되는 경우 스텝 S7의 활성화 상태로 천이하여 상술한 동작을 반복 수행하게 된다.
- <22> 그리고, 수동 절체되지 않는 경우 트윈이 비활성화 상태인지를 확인하여(스텝 S12), 트윈이 비활성화 상태가 아니면 자신은 스텝 S8의 비활성화 상태를 계속 유지하게 되고, 트윈도 비활성화 상태이면 즉, 이중 비활성화 상태가 되면 자신은 활성화 대기(Pending Active) 상태로 천이하게 된다(스텝 S13).
- <23> 그리고, 스텝 S4에서 트윈이 시동 상태인지를 확인한 결과 트윈이 시동 상태가 아닌 경우 해당 트윈이 활성화 상태인지를 확인하여(스텝 S14), 트윈이 활성화 상태이면 자신은 동기화 상태로 천이하게 된다(스텝 S15).
- <24> 해당 동기화 상태에서는 동기화가 완료되는지를 확인하여(스텝 S16), 동기화가 완료되지 않은 상태에서 트윈이 활성화 상태가 아니거나, 동기화가 완료된 상태에서 비활성화 대기 상태(Pending Standby)로 천이한 후에 트윈이 활성화 상태가 아니면(스텝 S17, S18), 자신은 스텝 S7의 활성화 상태로 천이하게 된다.
- <25> 그런데, 스텝 S17의 비활성화 대기 상태에서 트윈이 활성화 상태이면 자신은 스텝 S8의 비활성화 상태로 천이하게 된다.
- <26> 또한, 스텝 S13의 활성화 대기 상태에서 트윈이 활성화 상태인지를 확인하여(스텝 S19), 트윈이 활성화 상태이면 스텝 S8의 비활성화 상태로 천이하게 되고, 트윈이 활성화 상태가 아니면 스텝 S7의 활성화 상태로 천이하게 된다.
- <27> 그리고, 어떠한 상태에서든 알람(Alarm)이 발생하면 알람 상태로 천이하게 되고(스

탭 S20, S21), 해당 알람 상태가 해제되면(스텝 S22), 다시 스텝 S3의 시동 상태로 천이하게 된다.

<28> 한편, 트윈 알람 상태를 감지하면(스텝 S25), 자신이 활성화 상태인지를 확인하여(스텝 S26), 자신이 활성화 상태인 경우 활성화 상태를 계속적으로 유지하게 되고, 자신이 활성화 상태가 아닌 경우 스텝 S7의 활성화 상태로 천이하게 된다.

<29> 또한, 각 상태에서 어떠한 이유로 인해 시스템에 장애가 발생하거나 정지되면(스텝 S23), 정지(Halt) 상태로 천이하게 된다(스텝 S24).

<30> 상술한 바와 같이, 각 프로세서는 어떠한 이유로 인해 트윈이 활성화 상태가 아닌 것으로 확인되면, 비활성화 측은 활성화 대기 상태로 천이하게 되며, 해당 활성화 대기 상태에서 트윈이 활성화 상태로 복구되면 비활성화 상태로 천이하게 되고, 복구되지 않으면 활성화 상태로 천이하게 된다.

<31> 이때, 각 측이 동시에 비활성화 상태로 천이하게 되면 트윈이 비활성화 상태라는 것을 인식하는 순간 자신은 활성화 대기 상태로 천이하고, 해당 트윈이 활성화 대기 상태라는 것을 인식하면 자신은 다시 활성화 상태로 천이하게 되어 상태의 널뛰기(fluctuation, 변동) 현상이 발생하게 된다.

<32> 즉, 각 셀프는 트윈 감시 또는 자신의 상태를 트윈에게 통보하기 위해 주기적인 하트비트를 전송하고, 각 상태 천이시에도 하트비트를 트윈에게 전송하게 된다.

<33> 이때, 각 프로세서가 하트비트를 동시에 수신하게 되면 이중 활성화 또는 비활성화 상태로 천이하게 되어 해당 이중 활성화 상태와 비활성화 상태에서 널뛰기하는 현상이 발생하게 된다.

<34> 하지만, 이러한 현상을 방지하기 위해 상태 천이시 하트비트를 전송하지 않게 되면, 해당 상태 천이에 따른 보고가 지연되어 이중화 처리를 원활히 수행하지 못하는 문제점이 있다.

<35> 전술한 바와 같이, 종래에는 네트워크를 통해 프로세서를 소프트웨어적으로 이중화 처리하는 경우 해당 네트워크 장애 또는 네트워크 자원의 장애로부터 복구되는 시점에 각 프로세서가 하트비트를 동시에 수신하게 되면, 비정상적인 이중화 상태에 빠져 정상적인 이중화 처리를 수행하지 못하는 문제점이 있었다.

【발명이 이루고자 하는 기술적 과제】

<36> 본 발명은 전술한 바와 같은 문제점을 해결하기 위한 것으로 그 목적은, 이중화된 프로세서의 시동시에 난수 시드를 다르게 할당하여 난수 시간을 발생시키고, 해당 난수 시간을 사용하여 하트비트 전송 주기를 지속적으로 변경시켜 각 프로세서의 하트비트 수신 시각을 상이하게 함으로써, 비정상적인 이중화 상태의 발생을 방지할 수 있도록 하는데 있다.

<37> 본 발명의 다른 목적은, 이중화된 프로세서의 이중 활성화 상태 발생시 각 프로세서를 기설정된 상태로 천이하게 함으로써, 비정상적인 이중화 상태가 발생하더라도 신속하게 복구할 수 있도록 하는데 있다.

<38> 본 발명의 또 다른 목적은, 이중화 상태를 지속적으로 감지 및 제어하는 객체를 도입해서 프로세서의 이중화 상태 및 네트워크 자원을 감시하여 이중화 상태 천이를 제어

함으로써, 비정상적인 이중화 상태의 발생을 방지하고, 해당 비정상적인 이중화 상태가 발생하더라도 신속하게 복구할 수 있도록 하는데 있다.

#### 【발명의 구성 및 작용】

- <39>        상술한 바와 같은 목적을 달성하기 위한 본 발명의 특징은, 이중화된 프로세서의 이중화 상태 제어 방법에 있어서, 상기 각 프로세서에 난수 시드를 다르게 할당하여 난수 시간을 발생시키고, 상기 난수 시간을 사용하여 고정된 하트비트 전송 주기를 변경시켜 각 프로세서의 하트비트 수신 시각을 상이하게 하는 하트비트 난수 주기법을 포함하는데 있다. 그리고, 상기 하트비트 전송 주기는, 다음 하트비트 전송 시간에서 하트비트 수신 및 처리 시간을 감산하여 얻은 시간 이내로 변경시키는 것을 특징으로 하며, 상기 하트비트 수신 및 처리 시간은, 프로세서간의 하트비트 평균 도달 시간의 두 배에 평균 하트비트 처리 시간과 상태 천이 시간을 가산한 시간인 것을 특징으로 한다.
- <40>        본 발명의 다른 특징은, 이중화된 프로세서의 이중화 상태 제어 방법에 있어서, 상기 프로세서를 물리적인 측면에 따라 활성화 상태나 비활성화 상태로 각각 천이하도록 설정하고, 네트워크 관련 장애로부터 복구된 시점 또는 비정상적인 이중화 상태 발생시 각 프로세서가 설정된 상태로 각각 천이하게 하는 이중 활성화 협상법을 포함하는데 있다.
- <41>        본 발명의 또 다른 특징은, 이중화된 프로세서의 이중화 상태 제어 방법에 있어서, 상기 프로세서의 이중화 상태 및 네트워크 자원을 감시 및 제어하는 상태 감시/제어자를 도입하여 상기 프로세서의 이중화 상태 천이를 제어하게 하는 상태 감시/제어자 도입법

을 포함하는데 있다. 그리고, 상기 프로세서가 특정 상태에서 비정상적으로 장시간 동안 유지되는 경우 타이머와 자원을 할당받아 상기 프로세서를 상태 천이시키는 것을 특징으로 한다.

<42> 이하, 본 발명의 실시예를 첨부한 도면을 참조하여 상세하게 설명하면 다음과 같다.

<43> 본 발명은 소프트웨어적으로 이중화 처리를 수행하는 통신 시스템에서 이중화된 각 프로세서의 시동시에 난수(random number)를 발생시켜 고정된 하트비트의 전송 주기를 지속적으로 변경시키는 하트비트 난수 주기법과, 이중 활성화 상태가 되는 경우 지정된 상태로 각 프로세서가 상태 천이하게 하는 이중 활성화 협상(Dual Active Negotiation) 법과, 프로세서의 상태를 감시하고 제어하는 객체를 이용하는 상태 감시/제어자(State Monitor/Controller) 도입법을 제공한다.

<44> 첫번째, 하트비트 난수 주기법은 프로세서(A측, B측)의 주기적인 하트비트 전송 시간을 제어하기 위해 각 프로세서에 난수 시드(seed)를 다르게 할당하여 난수 시간을 발생시키고, 해당 난수 시간을 사용하여 고정된 하트비트 전송 주기를 지속적으로 변경시켜 각 프로세서의 하트비트 수신 시각을 상이하게 하는 방법으로써, 특히 각 프로세서가 네트워크 관련 장애로부터 복구되어 시동되는 시점에 상호 트윈 측으로 발신하는 한 번의 주기적인 하트비트 전송시 해당 하트비트의 전송 시간에 차이가 생겨 각 프로세서의 수신 시각이 상이하게 되고, 이에 따라 각 프로세서가 동시에 동일한 상태로 천이되는 비정상적인 이중화 상태 천이 즉, 이중 활성화 상태와 비활성화 상태의 널뛰기 현상을 방지할 수 있게 된다.

<45> 여기서, 각 프로세서의 고정된 하트비트 전송 주기를 ' $x$ ', 프로세서간의 하트비트 평균 도달 시간을 ' $a$ ', 프로세서의 평균 하트비트 처리 시간을 ' $b$ ', 상태 천이 시간을 ' $c$ '라고 가정하면, 각 프로세서는 트윈으로부터 하트비트를 수신하기 전에 이전 하트비트에 대한 처리를 완료하고 있어야 하며, 특정 시점에 해당 하트비트 전송로에는 최대한 개의 하트비트 메시지만이 존재해야 하므로, 고정된 하트비트 전송 주기는 ' $0 < (2a+b+c) < x$ '를 만족해야 한다.

<46> 그리고, 지속적으로 변경되는 하트비트 전송 주기를 ' $p$ '라고 하면, 해당 하트비트 전송 주기를 변경시킬 수 있는 최대 허용 범위는 다음 하트비트 전송 시간 이내이어야 함과 동시에 해당 하트비트 수신 및 처리 시간을 고려해야 하므로 ' $x < p < 2x - (2a+b+c)$ ' 즉, 다음 하트비트 전송 시간에서 하트비트 수신 및 처리 시간을 감산하여 얻은 시간 이내에서 변경되어야 한다.

<47> 이때, ' $a$ '는 평균값이므로 실제로 ' $a$ '보다 지연되는 경우를 고려하면 ' $x + (2a+b+c) \leq 2x - (2a+b+c)$ '를 만족해야 한다.

<48> 이에 따라, 고정된 하트비트 전송 주기 ' $x$ '와 변경된 하트비트 전송 주기 ' $p$ ' 사이에는 ' $2(2a+b+c) < x < p < 2x - (2a+b+c)$ '이 성립된다.

<49> 즉, 변경된 하트비트 전송 주기 ' $p$ '는 고정된 하트비트 전송 주기 이후부터 다음 하트비트 전송 주기 ' $2x$ '에서 하트비트 수신 및 처리 시간(프로세서간의 하트비트 평균 도달 시간의 두 배 ' $2a$ '에 해당 프로세서의 평균 하트비트 처리 시간 ' $b$ '와 상태 천이 시간 ' $c$ '를 가산한 시간 즉, ' $2a+b+c$ '를 감산하여 얻은 시간인 ' $2x - (2a+b+c)$ ' 이내에서 지속적으로 변경되어야 한다.

- <50> 두번째, 이중 활성화 협상법은 하트비트를 처리하고 상태 천이를 수행하는 루틴에서 각종 장애로부터 복구되어 시동되는 시점에 각 프로세서가 기설정된 상태로 천이하도록 하는 방법으로써, 각 프로세서가 고정된 물리적인 측면(A측, B측)에 따라 활성화 상태나 비활성화 상태로 각각 천이하도록 미리 설정하되, A측은 활성화 상태로, B 측은 비활성화 상태로 천이하도록 설정하거나 그 반대로 상태 천이하도록 설정할 수 있다.
- <51> 한편, 해당 이중 활성화 협상법에 따른 각 프로세서의 이중화 상태 제어 동작을 첨부한 도면 도 3을 참조하여 설명하면 다음과 같다.
- <52> 먼저, 각 프로세서는 비정상적인 이중화 상태인지를 확인하여(스텝 S31), 비정상적인 이중화 상태인 경우 즉, 자신이 활성화 상태이고 트윈도 활성화 상태라는 것을 인지하는 경우 자신이 A측인지를 확인한다(스텝 S32).
- <53> 이때, 자신이 A측이면 활성화 상태를 유지하게 되고(스텝 S33), 자신이 A측이 아니면 즉, 자신이 B측이면 즉시 비활성화 상태로 천이하으로써(스텝 S34), 비정상적인 이중 활성화 상태에서부터 신속하게 복구할 수 있게 된다.
- <54> 세번째, 상태 감시/제어자 도입법은 각 프로세서에 위치하여, 해당 프로세서의 이중화 상태를 지속적으로 감시하고 필요한 경우에 제어하는 제 3의 객체를 도입하는 방법으로써, 해당 프로세서의 이중화 상태 및 네트워크 자원을 감시하여 해당 프로세서의 이중화 상태 천이를 직접 제어하거나, 해당 네트워크 자원이 정상적이고 이중화 상태가 비정상적인 경우 상술한 하트비트 난수 주기법 또는 이중 활성화 협상법을 사용하여 제어할 수 있다.
- <55> 또한, 상태 천이시 특정 상태(예를 들면, 비활성화 대기 상태, 활성화 대기 상태,

시동 상태, 협상 상태, 동기화 상태 등)에서 비정상적으로 장시간 동안 유지되는지를 확인하여 강제적으로 상태 천이를 제어할 수 있게 된다.

<56> 이를 위해 새로운 타이머와 자원(resource)들을 시스템에서 하나 이상 할당받아야 하며, 링크 및 자신이 감시/제어하는 프로세서의 이중화와 관련된 자원에 대한 정보를 저장하고 있어야 한다.

<57> 한편, 본 발명에 따른 실시예는 상술한 것으로 한정되지 않고, 본 발명과 관련하여 통상의 지식을 가진자에게 자명한 범위내에서 여러 가지의 대안, 수정 및 변경하여 실시할 수 있는데, 예로서, 상술한 하트비트 난수 주기법과 이중 활성화 협상법 및 상태 감시/제어자 도입법을 각각 제공하거나, 두 가지 이상의 방법을 제공하여 프로세서의 이중화 상태를 제어할 수 있다.

#### 【발명의 효과】

<58> 이상과 같이, 본 발명은 이중화된 프로세서의 시동시에 난수 시드를 다르게 할당하여 난수 시간을 발생시키고, 해당 난수 시간을 사용하여 하트비트 전송 주기를 지속적으로 변경시켜 각 프로세서의 하트비트 수신 시각을 상이하게 함으로써, 비정상적인 이중화 상태의 발생을 방지할 수 있게 된다.

<59> 또한, 본 발명은 이중화된 프로세서의 이중 활성화 상태 발생시 각 프로세서를 기 설정된 상태로 천이하게 함으로써, 비정상적인 이중화 상태가 발생하더라도 신속하게 복구할 수 있게 된다.

<60> 그리고, 본 발명은 프로세서의 이중화 상태를 지속적으로 감지 및 제어하는 객체를



도입해서 해당 프로세서의 이중화 상태 및 네트워크 자원을 감시하여 이중화 상태 천이를 제어함으로써, 비정상적인 이중화 상태의 발생을 방지할 수 있게 되며, 해당 비정상적인 이중화 상태가 발생하더라도 신속하게 복구할 수 있게 된다.

**【특허청구범위】****【청구항 1】**

이중화된 프로세서의 이중화 상태 제어 방법에 있어서,

상기 각 프로세서에 난수 시드를 다르게 할당하여 난수 시간을 발생시키고, 상기 난수 시간을 사용하여 고정된 하트비트 전송 주기를 변경시켜 각 프로세서의 하트비트 수신 시각을 상이하게 하는 하트비트 난수 주기법을 포함하는 것을 특징으로 하는 이중화된 프로세서의 이중화 상태 제어 방법.

**【청구항 2】**

제 1항에 있어서,

상기 하트비트 전송 주기는, 다음 하트비트 전송 시간에서 하트비트 수신 및 처리 시간을 감산하여 얻은 시간 이내로 변경시키는 것을 특징으로 하는 이중화된 프로세서의 이중화 상태 제어 방법.

**【청구항 3】**

제 2항에 있어서,

상기 하트비트 수신 및 처리 시간은, 프로세서간의 하트비트 평균 도달 시간의 두 배에 평균 하트비트 처리 시간과 상태 천이 시간을 가산한 시간인 것을 특징으로 하는 이중화된 프로세서의 이중화 상태 제어 방법.

**【청구항 4】**

이중화된 프로세서의 이중화 상태 제어 방법에 있어서,

상기 프로세서를 물리적인 측면에 따라 활성화 상태나 비활성화 상태로 각각 천이하도록 설정하고, 네트워크 관련 장애로부터 복구된 시점 또는 비정상적인 이중화 상태 발생시 각 프로세서가 설정된 상태로 각각 천이하게 하는 이중 활성화 협상법을 포함하는 것을 특징으로 하는 이중화된 프로세서의 이중화 상태 제어 방법.

**【청구항 5】**

이중화된 프로세서의 이중화 상태 제어 방법에 있어서,

상기 프로세서의 이중화 상태 및 네트워크 자원을 감시 및 제어하는 상태 감시/제어자를 도입하여 상기 프로세서의 이중화 상태 천이를 제어하게 하는 상태 감시/제어자 도입법을 포함하는 것을 특징으로 하는 이중화된 프로세서의 이중화 상태 제어 방법.

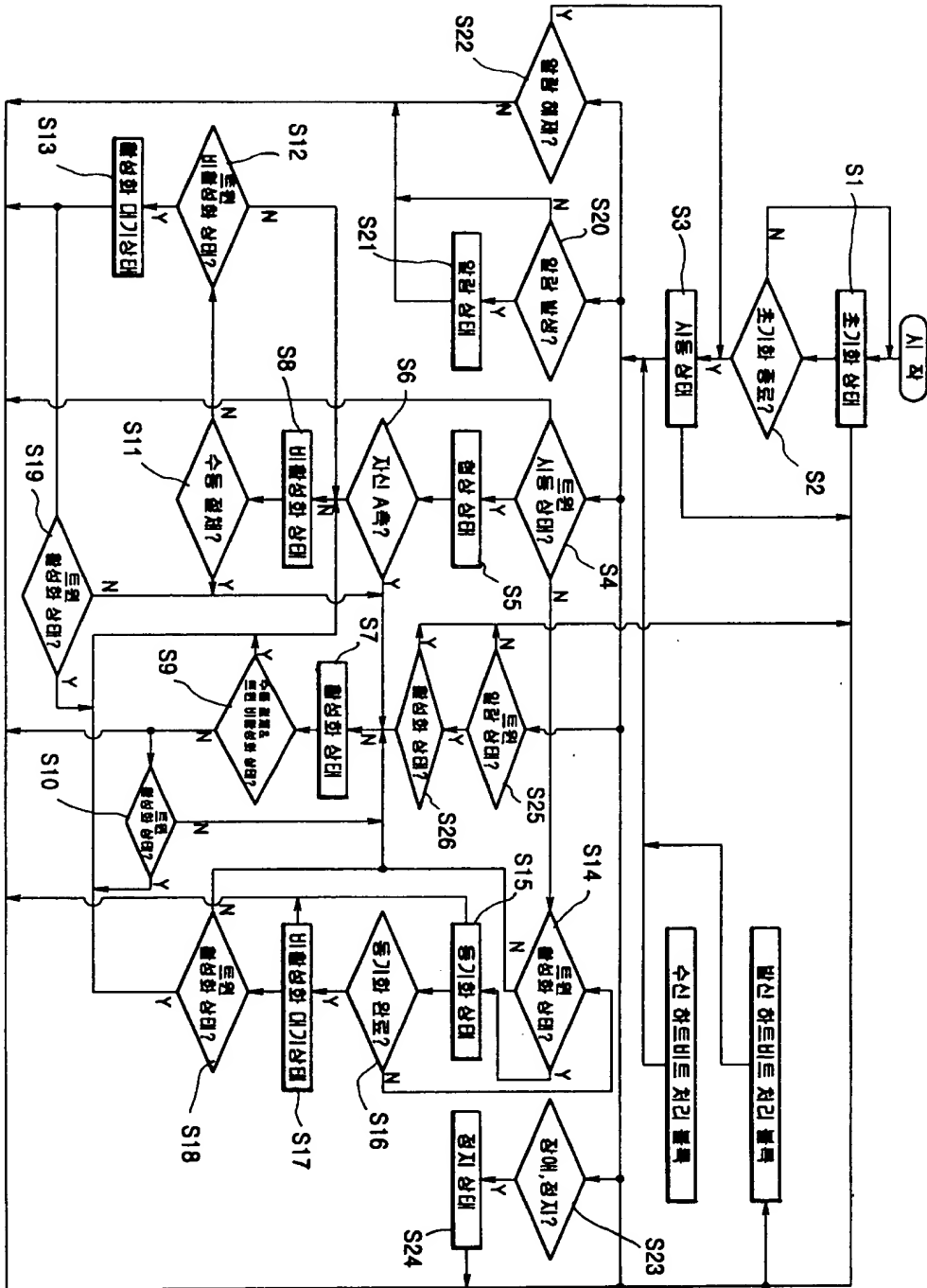
**【청구항 6】**

제 5항에 있어서,

상기 프로세서가 특정 상태에서 비정상적으로 장시간 동안 유지되는 경우 타이머와 자원을 할당받아 상기 프로세서를 상태 천이시키는 것을 특징으로 하는 이중화된 프로세서의 이중화 상태 제어 방법.



【도 2】



1019990028544

【도 3】

